**INDUSTRIAL INTERNSHIP REPORT ON**

**“QUIZ GAME”**

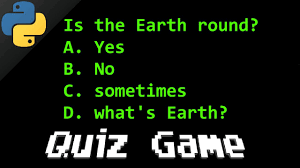
**PREPARED BY**

**THARUN GOPAL A**

|  |
| --- |
| *Executive Summary* |
| This report provides details of the Industrial Internship provided by upskill Campus and The IoT Academy in collaboration with Industrial Partner UniConverge Technologies Pvt Ltd (UCT).  This internship was focused on a project/problem statement provided by UCT. We had to finish the project including the report in 6 weeks’ time.  My project was Quiz Game.  This internship gave me a very good opportunity to get exposure to Industrial problems and design/implement solution for that. It was an overall great experience to have this internship. |

.**PROJECT REPORT:**

**QUIZ GAME PREPARATION USING PYTHON**

**INTROCUCTION**

The Quiz Game project will employ Python programming to develop a quiz game for consoles. Players will be presented with multiple-choice questions in the game, and their answers will be immediately evaluated. The scores will be kept track of and displayed after the game.

**OBJECTIVES**

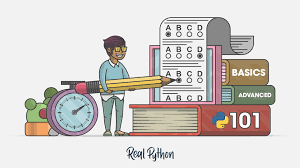
- Create a Python-based interactive quiz game.

- Compile a database of multiple-choice questions.

- Keep tabs on and compute the players' scores.

- Show the result and offer comments for each question.

- Permit participants to play the game more than once.



**TECHNOLOGIES USED**

Python: A programming language utilized in the creation of video games.

 - SQLite: Database management system for storing questions and answers.

**SYSTEM DESIGN**

The components of the quiz game project are as follows:

- Main Program Logic: Controls how the game progresses, how questions are presented, and how user input is handled.

Storage and retrieval of questions from a SQLite database is handled by database management.

- Score Tracking: Records each player's performance and determines the overall score.

- User Interface: Displays questions, options, and **remarks** to the players.

**IMPLEMENTATION OF STEPS**

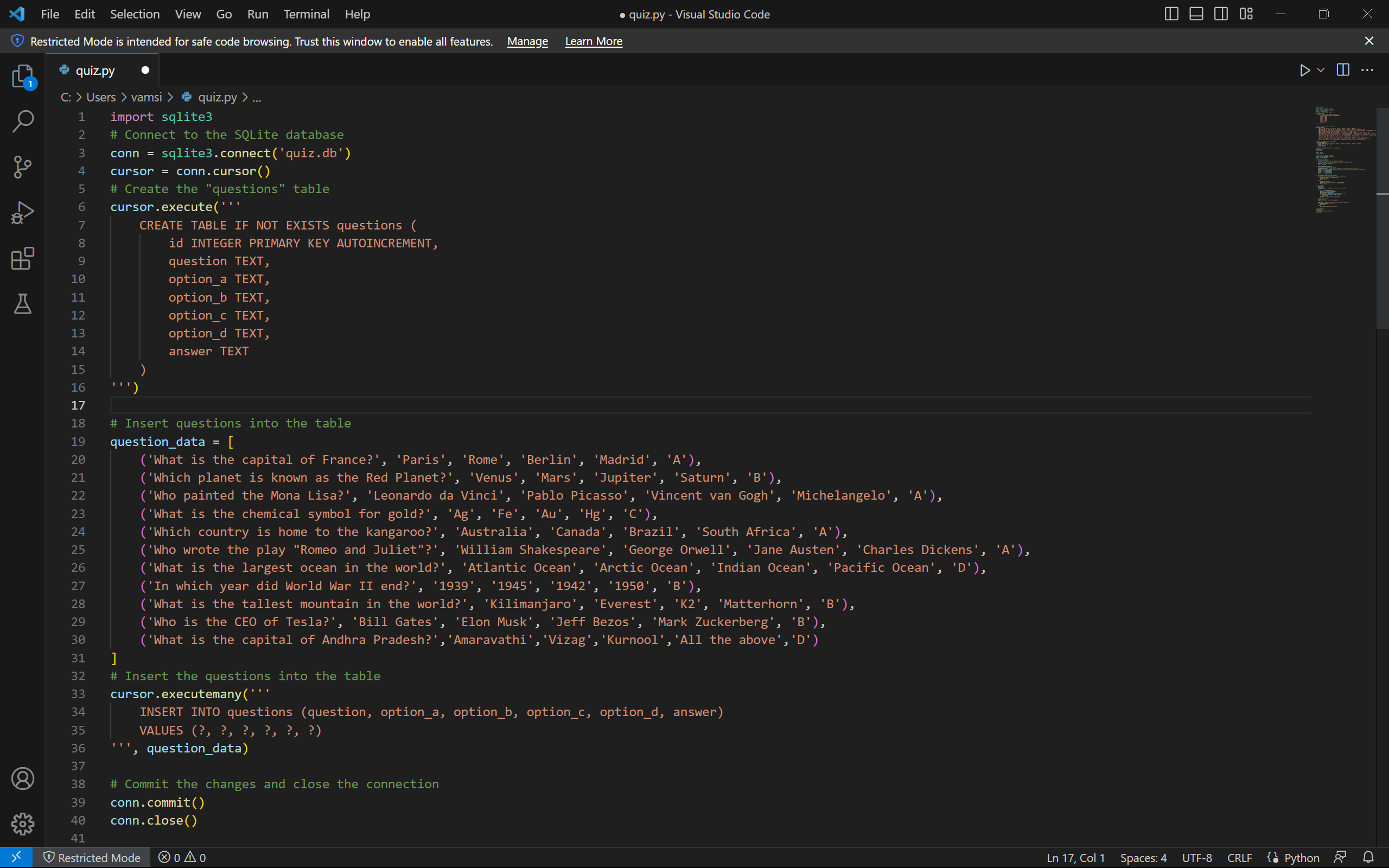
Step 1: Create an SQLite database to **keep** the questions and answers. Step 2: Write Python code **to hook up with** the database and retrieve questions.

Step 3: Develop **the sport** **good judgment** to **show** questions and **receive** **consumer** input.

Step 4: Calculate and **tune** the player's **rating** **primarily based totally** on their answers.

Step 5: Provide **instantaneously** **remarks** on **every** question.

Step 6: Display the **very last** **rating** and **provide** **the choice** to play again.

**SAMPLE CODE SNIPPET**

**CONCLUSION**

The Quiz Game project successfully implements a console-based quiz game using Python. It allows players to answer multiple-choice questions, calculates their score, and provides immediate feedback. The game can be expanded by adding more questions to the database and enhancing the user interface. This project enhances programming skills and knowledge of database management with Python.

By implementing this project, users can enjoy playing an engaging quiz game while also improving their Python programming skills.